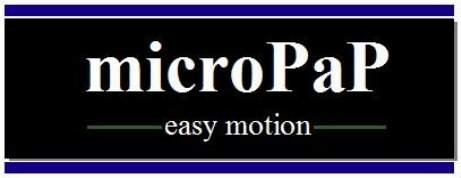

MicroPaP.dll Visual Basic Manual de usuario

ManMicroPaP dll Visual Basic
Rev. 1.4
23/10/2006
Autor: Ferran Sanabria
MicroPaP.com

ÍNDICE

1. INTRODUCCIÓN …….... 3

2. FUNCIONES IMPLEMENTADAS EN LA LIBRERIA 3

3. DESCRIPCIÓN DE LAS FUNCIONES 3

- IniComunicationsPaP..... 3
- EnviarTrama..... 4
- TramaEnviada..... 7
- TramaRecibida..... 8
- EndComunicationsPaP..... 8

4. EJEMPLO APLICACIÓN..... 9

1. INTRODUCCIÓN

Esta documentación explica como utilizar la libreria de vinculos dinámicos “micropap.dll” a través de Microsoft Visual Basic TM.

2. FUNCIONES IMPLEMENTADAS EN LA LIBRERIA

Function *IniComunicationsPaP* (ByVal port_arg As String, ByVal rate_arg As Integer, ByVal parity_arg As Byte, ByVal ByteSizechar As Byte) As Integer

Function *EnviarTrama* (ByVal identificador As Byte, ByVal N_Pasos_Totals As Integer, ByVal Sentit_Gir As Byte, ByVal N_Pasos_Aceleracio As Byte, ByVal N_Pasos_Deceleracio As Byte, ByVal Velocitat_Max As Byte, ByVal Velocitat_Min As Byte, ByVal MicroPasos As Byte, ByVal CorrentStop As Byte, ByVal SearchHome As Byte, ByVal ContinuousMov As Byte, ByVal Prescaler As Byte) As Integer

Function *TramaRecibida* (ByVal BufferRxUser As String) As Integer

Function *TramaEnviada* (ByVal BufferTxUser As String) As Integer

Function *EndComunicationsPaP* () As Integer

3. DESCRIPCIÓN DE LAS FUNCIONES

IniComunicationsPaP

Explicación

Esta función inicializa el puerto de comunicaciones y lo deja preparado para su uso. Se debe llamar a esta función para poder comunicar a traves del puerto RS-232.

Declaración

IniComunicationsPaP Lib "micropap_dll.dll" (ByVal port_arg As String, ByVal rate_arg As Integer, ByVal parity_arg As Byte, ByVal ByteSizechar As Byte) As Integer

Parametros

ByVal port_arg

String. Especifica el nombre del puerto de comunicaciones a utilizar.

Integer rate_arg

Entero. Especifica la velocidad de comunicaciones.

byte_parity parity_arg

0 : NONE , 1 : spODD , 2 : spEVEN

Paridad. Especifica la paridad a utilizar en las comunicaciones

byte ByteSizechar

byte. Especifica el tamaño del byte de datos.

Uso

char error;

Result = IniComunicationsPaP("COM1", 19200, 0, 8)

Retorna

0 Ningún error
-1 Error al conectar el dispositivo de comunicaciones
-2 No se ha podido crear el dispositivo de comunicaciones

EnviarTrama

Explicación

Esta rutina envia una trama a traves del puerto de comunicaciones para ejecutar un determinado comando.

Declaración

Function EnviarTrama (ByVal identificador As Byte, ByVal N_Pasos_Totals As Integer, ByVal Sentit_Gir As Byte, ByVal N_Pasos_Aceleracio As Byte, ByVal N_Pasos_Deceleracio As Byte, ByVal Velocitat_Max As Byte, ByVal Velocitat_Min As Byte, ByVal MicroPasos As Byte, ByVal CorrentStop As Byte, ByVal SearchHome As Byte, ByVal ContinuousMov As Byte, ByVal Prescaler As Byte) As Integer

Parametros

Byte identificador

Byte. Especifica el identificador del motor. El primer byte de la trama a enviar a la controladora es el que identifica el motor al que va dirigida la trama. En el caso de la controladora MPAP-1AXV1 al ser una conexión punto a punto este byte es fijo y el valor que debe tener es 0x30 . Este podrá tener diferentes valores si se trabaja a través de la controladora de 10 motores MPAP-HOST-6AXV1 (0x30..0x39).

integer N_Pasos_Totals

Entero. Especifica el número de pasos a efectuar en regimen estable.

Byte Sentit_Gir

Byte. Especifica el sentido de giro. Se deberá enviar el código 0x82 para girar en sentido de las agujas del reloj (CW, clock wise) o 0x76 para girar en sentido contrario al de las agujas del reloj (CCW, counter clock wise).

Byte N_Pasos_Aceleracio

Byte. Especifica el número de pasos que realizará el motor en la rampa de aceleración para alcanzar la velocidad de regimen. El valor está limitado entre 0 y 99 (0x00+0x30 .. 0x63). Se debe sumar 0x30 al valor enviado.

Byte N_Pasos_Deceleracio

Byte. Especifica el número de pasos que realizará el motor en la rampa de deceleración para alcanzar la velocidad mínima antes de la parada . El valor está limitado entre 0 y 99. (0x00+0x30 .. 0x63+0x30). Se debe sumar 0x30 al valor enviado.

Byte Velocitat_Max

Caracter. Especifica la velocidad máxima a la que se moverá el motor en funcionamiento estable o de régimen. El valor está limitado entre 0 y 200 (0x00+0x30 .. 0xC8+0x30). Se debe sumar 0x30 al valor enviado

Byte Velocitat_Min

Byte. Especifica la velocidad a la se moverá el motor antes de pararse. El valor está limitado entre 0 y 200 (0x00+0x30 .. 0xC8+0x30). Se debe sumar 0x30 al valor enviado

Byte MicroPasos

Byte. Especifica el número de micropasos que se pueden realizar entre pasos. Los valores que puede tomar son :

- 0x00 : pasos completos
- 0x31 : 1/2 pasos
- 0x32 : 8 micropasos entre pasos
- 0x33 : 16 micropasos entre pasos

- 0x34 : 32 micropasos entre pasos
- 0x35 : 64 micropasos entre pasos

Byte CorrentStop

Byte. El byte 10 indica la corriente que se desea que se le suministre al motor cuando este parado. Esta corriente (indicada en mA) permitirá mantener el par al motor en las paradas. El valor está limitado entre 0 y 63 (0x00+0x30 .. 0x3F+0x30). Se debe sumar 0x30 al valor enviado

Byte SearchHome

Byte. Búsqueda de los finales de carrera FC1 o FC2

- 0x30 = Testea FC durante los movimientos.
- 0x31 = Testea FC1 girando derecha
- 0x32 = Testea FC2 girando izquierda
- 0x33 = No testea los FC durante los movimientos.

Byte ContinuousMov (modo infinito)

Byte. Especifica si se quiere realizar un movimiento continuo sin atender al número de pasos la trama. Los valores que puede tomar son :

- 0x30 : no movimiento en continuo. Unicamente tiene sentido si se está ejecutando un movimiento en continuo. Para el movimiento en continuo.
- 0x31 : Lanza el movimiento en continuo.

Byte Prescaler

Byte. Especifica el prescaler de velocidad. Los valores que puede tomar son :

- 0x30 = 1/2 (máxima velocidad)
- 0x31 = 1/4
- 0x32 = 1/8
- 0x33 = 1/16
- 0x34 = 1/32

- 0x35 = 1/64
- 0x36 = 1/128 (valor por defecto)
- 0x37 = 1/256 (mínima velocidad)

Uso

EnviarTrama (&H32, 400, &H52, &H30, &H30, &HFA+&H30, &H30, &H33, &H30, &H33, &H30, &H35);

Retorna

Retorna 0 si el comando se ha ejecutado correctamente . Para saber el resultado del envio se debe consultar el buffer de transmisión y recepción.

En el caso de error en el rango de alguno de los parámetros retorna el número de orden del parámetro en valor negativo. (-1 : error en el rango del identificador, -2 : error en el rango del número de pasos totales,….)

TramaEnviada

Explicación

Esta función consulta la información enviada por el puerto de comunicaciones y retorna los bytes enviados a partir del String que se le pasa como parámetro.

Declaración

Function TramaEnviada (ByVal BufferTxUser As String) As Integer

Parametros

String BufferTxUser

String. String a partir del cual se retornarán los bytes que hayan sido transmitidos por el puerto de comunicaciones.

Uso

Dim BufferTx as String

Dim IndTx as Integer

Indtx = TramaRecibida(BufferTx)

Retorna

Retorna el número de bytes que han sido enviados por el puerto de comunicaciones

TramaRecibida

Explicación

Esta función consulta la información recibida por el puerto de comunicaciones y retorna los bytes recibidos a partir del String que se le pasa como parámetro.

Declaración

Function TramaRecibida (ByVal BufferRxUser As String) As Integer

Parametros

String BufferRxUser

String. String a partir del cual se retornarán los bytes que hayan sido recibidos por el puerto de comunicaciones.

Uso

```
Dim BufferRx as String
Dim IndRx as Integer
IndRx = TramaRecibida(BufferRx)
```

Retorna

Retorna el número de bytes que han sido recibidos por el puerto de comunicaciones

EndComunicationsPaP

Explicación

Finaliza las comunicaciones RS-232

Declaración

Function EndComunicationsPaP () As Integer

Parametros

No tiene

Uso

Result = EndComunicationsPaP

Retorna

No retorna nada.

4. EJEMPLO APLICACIÓN

```
Dim Result, IndTx, IndRx As Integer
Dim BufferTx As String
Dim BufferRx As String
Private Sub Enviar_Click()
Dim b(1 To 100) As Byte
' Inicializar los strings de BufferTx y BufferRx.
' Esta inicialización es obligatoria, si se omite
' no pasa bien los parámetros en las rutinas TramaEnviada y TramaRecibioda
BufferTx = " "
BufferRx = " "
' Envia una trama :
'  identificador = 0x32;
'  N_Pasos_Totals = 400 pasos;
'  Sentit_Gir = 0x52; 82 en decimal
'  N_Pasos_Aceleracio = 0x00;
'  N_Pasos_Deceleracio = 0x00;
'  Velocitat_Max = 0xfa; 250 en decimal
 '  Velocitat_Min = 0x00;
 '  MicroPasos = 0x03 : 32 micropasos
```

```
'  CorrentStop = 0x00;
 '  SearchHome = 0x03; No testea FC durante los movimientos
'  ContinuousMov = 0x00; Para el movimiento continuo
'  Prescaler = 0x05 ;
Result =EnviarTrama (&H32, 400, &H52, &H30, &H30, &HFA+&H30, &H30, &H33, &H30, &H33, &H30,
&H35);
' Carga el buffer de transmision interno a partir de &BufferTx[0]
IndTx = TramaEnviada(BufferTx)
' Se recoge el resultado
For i = 1 To IndTx * 2
        b(i) = AscB(MidB(BufferTx, i))
Next i
' el paso de parametros a la dll genera un byte adicional en ' cada parametro que despreciamos (i=i+1)
For i = 1 To IndTx * 2
        List1.AddItem (Hex$(b(i))) i =i + 1
Next i
' Carga el buffer de recepcion interno a partir de &BufferRx[0]
IndRx = TramaRecibida(BufferRx)
' Se recoge el resultado
For i = 1 To IndRx * 2
        b(i) = AscB(MidB(BufferRx, i))
Next i
' el paso de parametros a la dll genera un byte adicional en ' cada parametro que despreciamos (i=i+1)
For i = 1 To IndRx * 2
        List2.AddItem (Hex$(b(i))) i =i + 1
Next i
End Sub
```

```
Private Sub Form_Load()
Result = IniComunicationsPaP("COM1", 19200, 0, 8)
End Sub

Private Sub Form_Terminate()
' Libera el canal de comunicación
Result = EndComunicationsPaP
End Sub

Private Sub Form_Unload(Cancel As Integer)
Result = EndComunicationsPaP
End Sub

Private Sub Salir_Click()
' Libera el canal de comunicación
Result = EndComunicationsPaP End End Sub Fichero
```

Module1.bas

```
Public Declare Function IniComunicationsPaP Lib "micropap_dll.dll" (ByVal port_arg As String, ByVal rate_arg
As Integer, ByVal parity_arg As Byte, ByVal ByteSizechar As Byte) As Integer

Public Declare Function EnviarTrama Lib "micropap_dll.dll" (ByVal identificador As Byte, ByVal
N_Pasos_Totals As Integer, ByVal Sentit_Gir As Byte, ByVal N_Pasos_Aceleracio As Byte, ByVal
N_Pasos_Deceleracio As Byte, ByVal Velocitat_Max As Byte, ByVal Velocitat_Min As Byte, ByVal MicroPasos
As Byte, ByVal CorrentStop As Byte, ByVal SearchHome As Byte, ByVal ContinuousMov As Byte, ByVal
Prescaler As Byte) As Integer
```

```
Public Declare Function TramaRecibida Lib "micropap_dll.dll" (ByVal BufferRxUser As String) As Integer

Public Declare Function TramaEnviada Lib "micropap_dll.dll" (ByVal BufferTxUser As String) As Integer

Public Declare Function EndComunicationsPaP Lib "micropap_dll.dll" () As Integer
```